Outline

• **Introduction:**
  *What is a Packet Switch?*

• **Packet Lookup and Classification:**
  *Where does a packet go next?*

• **Switching Fabrics:**
  *How does the packet get there?*

• **Output Queuing:**
  *What happens before the packet goes out?*
Introduction

What is a Packet Switch?

• Basic Architectural Components
• Some Example Packet Switches
Basic Architectural Components

Datapath: per-packet processing

Control

Routing

Congestion Control

Admission Control

Policing

Switching

Output Scheduling

Reservation
Basic Architectural Components

Datapath: per-packet processing

1. Forwarding Table
   - Forwarding Decision

2. Interconnect

3. Output Scheduling

Copyright 1999. All Rights Reserved
Where high performance packet switches are used

- Carrier Class Core Router
- ATM Switch
- Frame Relay Switch

The Internet Core

Edge Router

Enterprise WAN access & Enterprise Campus Switch
Introduction

What is a Packet Switch?

• Basic Architectural Components
• Some Example Packet Switches
Ethernet Switch

Routing is done by caching what addresses are on what ports.

- Lookup frame DA in forwarding table.
  - If known, forward to correct port.
  - If unknown, broadcast to all ports.
- Learn SA of incoming frame.
- Forward frame to outgoing interface.
- Transmit frame onto link.
Routing algorithm initializes forwarding table.
• Match packet DA to a prefix in the forwarding table.
  – If match exists, forward to correct port.
  – If no match, drop packet.
• Decrement TTL, update header Cksum.
• Forward packet to outgoing interface.
• Transmit packet onto link.
Routing algorithm initializes fixed-size routing table (VC table).

- Lookup cell VCI/VPI in VC table.
- Replace old VCI/VPI with new.
- Forward cell to outgoing interface.
- Transmit cell onto link.
Outline

- **Introduction:**
  *What is a Packet Switch?*

- **Packet Lookup and Classification:**
  *Where does a packet go next?*

- **Switching Fabrics:**
  *How does the packet get there?*

- **Output Queuing:**
  *What happens before the packet goes out?*
Basic Architectural Components

Datapath: per-packet processing
Forwarding Decisions

- Bridges and Ethernet switches
  - Associative Lookup
  - Hashing
- IP Routers
  - Caching
  - CIDR
- ATM and MPLS switches
  - Direct Lookup
- Packet Classification
# Bridges and Ethernet Switches

## Associative Lookups

<table>
<thead>
<tr>
<th>Network Address</th>
<th>Associated Data</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Search Data**

48

**Advantages:**
- Simple

**Disadvantages:**
- Slow
- High Power
- Small
- Expensive

** Associative Memory or CAM **

** Hit? **

** Address **

\( \log_2 N \)
Lookups Using Hashing

An example

Hashing Function
CRC-16

Memory

Search Data

Associated Data
Hit?
Address
$\log_2 N$

Indexed lists

Copyright 1999. All Rights Reserved
Forwarding Decisions

- Bridges and Ethernet switches
  - Associative Lookup
  - Hashing
- IP Routers
  - Caching
  - CIDR
- ATM and MPLS switches
  - Direct Lookup
- Packet Classification
IP Routers

Class-based addresses

IP Address Space

| Class A | Class B | Class C | D |

Routing Table:

Exact Match

212.17.9.4

Class A

Class B

Class C
IP Routers

CIDR

Class-based:

Classless:

128.9.0.0

128.9.16.14

142.12/19

Copyright 1999. All Rights Reserved
Most specific route = “longest matching prefix”
IP Routers

*Metrics for Lookups*

<table>
<thead>
<tr>
<th>Prefix</th>
<th>Port</th>
</tr>
</thead>
<tbody>
<tr>
<td>65/8</td>
<td>3</td>
</tr>
<tr>
<td>128.9/16/20</td>
<td>2</td>
</tr>
<tr>
<td>128.9/19/24</td>
<td>7</td>
</tr>
<tr>
<td>128.9.16/20</td>
<td>5</td>
</tr>
<tr>
<td>128.9.25/24</td>
<td>10</td>
</tr>
<tr>
<td>128.9.16/24</td>
<td>2</td>
</tr>
<tr>
<td>128.9.176/20</td>
<td>1</td>
</tr>
<tr>
<td>142.12/19</td>
<td>3</td>
</tr>
</tbody>
</table>

- Lookup time
- Storage space
- Update time
- Preprocessing time
IP Router

**Lookup**

IPv4 unicast destination address based lookup

- **Forwarding Table**
  - Destination | Next Hop
  - ---- | ----
  - ---- | ----
  - ---- | ----
  - ---- | ----

- **Next Hop Computation**
- **Forwarding Engine**
- **Incoming Packet**
- **Dstn Addr**
- **Next Hop**

Copyright 1999. All Rights Reserved
Trees and Tries

Binary Search Tree

Binary Search Trie

\[ \log_2 N \]

\[ N \text{ entries} \]
Trees and Tries

*Multiway tries*

16-ary Search Trie

```
  0000, ptr
  /       \
0000, 0  1111, ptr
   /       \
  1111, ptr
```

```
  0000, 0
  /       \
1111, ptr
   /       \
  0000, 0
     /       \
1111, ptr
```

Copyright 1999. All Rights Reserved
Forwarding Decisions

- Bridges and Ethernet switches
  - Associative Lookup
  - Hashing
- IP Routers
  - Caching
  - CIDR
- ATM and MPLS switches
  - Direct Lookup
- Packet Classification
ATM and MPLS Switches

Direct Lookup

VCI

Address

Memory

Data

(Port, VCI)
Outline

• Introduction:
  *What is a Packet Switch?*

• Packet Lookup and Classification:
  *Where does a packet go next?*

• Switching Fabrics:
  *How does the packet get there?*

• Output Queuing:
  *What happens before the packet goes out?*
Switching Fabrics

• Evolution of IP Routers
  • Output and Input Queueing
  • Output Queueing
  • Input Queueing
  • Other non-blocking fabrics
First-Generation IP Routers

Shared Backplane

CPU

Memory

Line Interface

CPU

Buffer Memory

DMA

Line Interface

MAC

DMA

Line Interface

MAC

DMA

Line Interface

MAC
Second-Generation IP Routers
Third-Generation Switches/Routers

- CPU Card
- Local Buffer Memory
- MAC
- Line Card
- CPU Card
- Line Card
- Local Buffer Memory
- MAC
Fourth-Generation Switches/Routers
Clustering and Multistage
Switching Fabrics

- Evolution of IP Routers

- Output and Input Queueing
  - Output Queueing
  - Input Queueing

- Other non-blocking fabrics
Basic Architectural Components

Datapath: per-packet processing
Interconnects

Two basic techniques

Input Queueing

Usually a non-blocking switch fabric (e.g. crossbar)

Output Queueing

Usually a fast bus
Interconnects
Output Queueing

Individual Output Queues

Centralized Shared Memory

Memory b/w = (N+1).R

Memory b/w = 2N.R
Output Queueing

How fast can we make centralized shared memory?

- 5ns per memory operation
- Two memory operations per packet
- Therefore, up to 160Gb/s
- In practice, closer to 80Gb/s
Interconnects

Input Queueing with Crossbar

Data In

Data Out

configuration

Memory b/w = 2R

Scheduler

Copyright 1999. All Rights Reserved
Input Queueing

Head of Line Blocking

Copyright 1999. All Rights Reserved
Head of Line Blocking
Input Queueing

*Virtual output queues*
Input Queues

Virtual Output Queues

[Diagram of input queues and virtual output queues]

[Graph showing delay vs. load]
Input Queueing

Memory b/w = 2R

Can be quite complex!
Switching Fabrics

- Evolution of IP Routers
- Output and Input Queueing
  - Output Queueing
  - Input Queueing
- Other non-blocking fabrics
Other Non-Blocking Fabrics

*Clos Network*
Other Non-Blocking Fabrics

*Clos Network*

Expansion factor required = 2-1/N  (but still blocking for multicast)
Other Non-Blocking Fabrics

Self-Routing Networks
The Non-blocking Batcher Banyan Network

**Batcher Sorter**

**Self-Routing Network**

- Fabric can be used as scheduler.
- Batcher-Banyan network is blocking for multicast.